



ԳՆԱՀԱՏՄԱՆ ԵՎ ԹԵՍՏԱՎՈՐՄԱՆ ԿԵՆՏՐՈՆ

ՆԱԽՆԱԿԱՆ ԵՎ ՄԻՋԻՆ ՄԱՍՆԱԳԻՏԱԿԱՆ  
ՈՒՍՀԱՍՏԱՏՈՒԹՅԱՆ ՄԱՆԿԱՎԱՐԺԻ ԱՏԵՍՏԱՎՈՐՈՒՄ

2025

ԾՐԱԳՐԱՎՈՐՄԱՆ ՀԻՄՈՒՆՔՆԵՐ

ԹԵՍՏ 4

ԽՄԲԻ ՀԱՄԱՐԸ

ՆՍՏԱՐԱՆԻ ՀԱՄԱՐԸ

*Հարգելի՛ ուսուցիչ*

Խորհուրդ ենք տալիս առաջադրանքները կատարել ըստ հերթականության:  
Ուշադիր կարդացե՛ք յուրաքանչյուր առաջադրանք: Եթե չի հաջողվում որևէ առաջադրանքի  
անմիջապես պատասխանել, ժամանակը խնայելու նպատակով կարող եք դրան  
անդրադառնալ ավելի ուշ:

*Թեստ-գրքույկի էջերի դատարկ մասերը կարող եք օգտագործել սնագրության  
համար:*

1

Ստորև նշված որ համակարգն է ամպային օպերացիոն հարթակ՝ նախատեսված ինտերնետում ծրագրերի և ծառայությունների տեղակայման ու կառավարման համար:

- 1) Linux
- 2) Azure
- 3) Windows Server
- 4) Windows 11

2

Ի՞նչ ընդհանրություն կա .docx և .xlsx ֆայլերի կառուցվածքի մեջ:

- 1) Երկու ձևաչափերն էլ սովորական տեքստային ֆայլեր են
- 2) Երկու ձևաչափերն էլ պահում են տվյալները միայն երկուական կոդով՝ առանց տեքստային նկարագրության
- 3) Երկու ձևաչափերն էլ անհամատեղելի են ամպային ծառայությունների հետ
- 4) Երկու ձևաչափերն էլ ZIP-արխիվներ են, որոնք պարունակում են XML ֆայլեր՝ տվյալների և փաստաթղթի կառուցվածքի պահպանման համար

3

Ո՞ր պնդումն է ճիշտ նկարագրում Microsoft Office-ի Clipboard-ի հնարավորությունները:

- 1) Clipboard-ը չի պահում պատկերներ, միայն տեքստ
- 2) Clipboard-ը ինքնաբերաբար ջնջում է օբյեկտները յուրաքանչյուր 5 րոպեն մեկ
- 3) Clipboard-ը կարող է միաժամանակ պահել մինչև 24 օբյեկտ
- 4) Clipboard-ը միջոցով հնարավոր չէ պատճենել աղյուսակները Excel-ից դեպի Word

4

Ի՞նչ հնարավորություն է տալիս Microsoft Office-ի արագ հասանելիության գործիքագոտու (Quick Access Toolbar) և ժապավենի (Ribbon) կարգավորումը:

- 1) Ավելացնել, հեռացնել և վերադասավորել հրամանները՝ օգտվողի հարմարավետ աշխատանքի համար
- 2) Փոխել Word և Excel ֆայլերի ձևաչափը
- 3) Կառավարել օգտվողների մուտքը փաստաթղթերին
- 4) Փոխել Office-ի ինտերֆեյսի լեզուն

5

Ի՞նչ է կատարում Transcribe գործիքը Microsoft Word-ում (Office 365-ում):

- 1) Ստեղծում է փաստաթղթի ձևանմուշ
- 2) Ձայնային ֆայլերը վերածում է տեքստի
- 3) Թարգմանում է տեքստը մեկ լեզվից մյուսը
- 4) Ավտոմատ ուղղում է տեքստային սխալները

6

Ի՞նչ պետք է անել, որպեսզի Microsoft Word կամ Excel ծրագրերում ավելացվի նոր ֆոնտ և այն հայտնվի ֆոնտերի ցուցակում:

- 1) Ֆոնտը պետք է ավելացնել Clipboard-ի միջոցով
- 2) Ֆոնտը պետք է ներբեռնել և տեղադրել օպերացիոն համակարգի Fonts թղթապանակում
- 3) Ֆոնտերի ցուցակում նոր ֆոնտ ավելացնելու համար պետք է կիրառել MS Office տեղադրման փաթեթը
- 4) Ֆոնտը պետք է ներմուծել անհրաժեշտ փաստաթղթի մեջ

7

Երբ տվյալները փոփոխվում են սկզբնական աղյուսակում (source data), ի՞նչ քայլ է անհրաժեշտ կատարել Pivot Table-ում, որպեսզի ցուցադրվեն թարմացված արդյունքները:

- 1) Պետք է ընտրել Pivot Table-ը և կատարել Refresh հրամանը
- 2) Պետք է կրկին ստեղծել Pivot Table-ը զրոյից
- 3) Պետք է ֆիլտրերը անջատել և նորից միացնել
- 4) Փոփոխությունները ինքնաբերաբար արտացոլվում են Pivot Table-ում առանց որևէ գործողության

8

Ի՞նչն է ամենաճշգրիտ կերպով բնորոշում թոքենիզատորի (tokenizer) դերը LLM (Large Language Model) համակարգերում:

- 1) Թոքենիզատորը փոխակերպում է տեքստը թվային ներկայացումների, որոնք հասկանալի են մոդելին:
- 2) Թոքենիզատորը պատասխանատու է մոդելի քաշերի վերականգման և կարգաբերման համար:
- 3) Թոքենիզատորը կառավարում է տեքստի գեներացումը և "temperature" պարամետրը:
- 4) Թոքենիզատորը որոշում է նեյրոնային ցանցի կառուցվածքը և շերտերի քանակը:

1.

```
from openai import OpenAI
client = OpenAI()
response = client.chat.completions.create(
    model="gpt-4o",
    messages=[{"role": "user", "content": "Hello!"}])
print(response.choices[0].message["content"])
```

2.

```
import openai
response = openai.Model.generate(
    name="gpt-4",
    prompt="Hello!")
print(response.output)
```

3.

```
from openai import GPT
client=chat
model=noName
GPT.run("Hello!")
print(response.output)
```

4.

```
import openai
response = openai.ChatCompletion.create(
    model="gpt-6",
    input="Hello!")
print(response.text)
```

10

Ի՞նչ արժեք կարտածի հետևյալ Python կոդը.

```
a = 40
b = 2
result = a >> b
print(result)
```

- 1) 20
- 2) 30
- 3) 40
- 4) 10

11

Ի՞նչ հիմնական նպատակ ունի LOOKUP ֆունկցիան Excel-ում:

- 1) Աղյուսակում որոշակի արժեքի որոնում
- 2) Թվերի միջին արժեքի հաշվարկում
- 3) Աղյուսակի տողերի և սյուների փոխանակում
- 4) Տեքստերի համատեղում և միացում

12

Ի՞նչ արժեք կստացվի հետևյալ բանաձևից՝

=COUNT(A1:A5), եթե բջիջների արժեքներն են՝  
A1=12, A2="Hello", A3=7, A4=15, A5 դատարկ է

- 1) 3
- 2) 4
- 3) 2
- 4) 5

13

Եթե B1:B5 տիրույթում արժեքներն են՝

B1=10, B2 դատարկ է, B3 դատարկ է, B4=25, B5="Data"

Ի՞նչ արժեք կտա բանաձևը՝ =COUNTBLANK(B1:B5)

- 1) 2
- 2) 3
- 3) 4
- 4) 1

14

Նշվածներից ո՞ր տվյալների տիպն է չփոխվող (immutable) Python-ում:

- 1) Dictionary
- 2) Tuple
- 3) Set
- 4) List

15

Ի՞նչ կարտածի հետևյալ Python կոդը.

```
a = [1, 2, 3]
b = [1, 2, 3]
print(a == b)
print(a is b)
```

- 1) True  
False
- 2) False  
True
- 3) False  
False
- 4) True  
True

16

Ի՞նչ կարտածի հետևյալ Python կոդը.

```
x = "data"
y = "data"
z = ["data"]
print(x == y)
print(x is z)
```

- 1) True  
True
- 2) False  
True
- 3) False  
False
- 4) True  
False

17

Ի՞նչ կարտաձի հետևյալ Python կոդը.

```
x = "data"  
y = "data"  
z = ("data")  
print(x == y)  
print(x == z)
```

- 1) True  
False
- 2) False  
True
- 3) False  
False
- 4) True  
True

18

Ի՞նչ կարտաձի հետևյալ Python կոդը.

```
*a, b, c = 10, 20, 30, 40, 50  
print(a)
```

- 1) [30, 40]
- 2) 10
- 3) [20, 30, 40, 50]
- 4) [10, 20, 30]

19

Ի՞նչ կարտաձի հետևյալ Python կոդը.

```
t1 = (5, 10, 15)  
t2 = (5, 8, 20)  
print(t1 > t2)
```

- 1) False
- 2) TypeError
- 3) FatalError
- 4) True

20

Ի՞նչ կարտածի հետևյալ Python կոդը (— նշանակում է բացատ).

```
def calc(x):  
    x = x + 5  
    — y = x * 2  
    — return y  
print(calc(3))
```

- 1) IndentationError
- 2) NameError
- 3) FatalError
- 4) SyntaxError

21

Python-ի tuple-ների վերաբերյալ տրված պնդումներից 3-ը ճիշտ են: Նշել դրանց համարները: *(3 միավոր)*

1. Tuple-ը կարելի է փոխարկել List-ի՝ list() ֆունկցիայի միջոցով
2. Tuple-ը կարելի է ընդլայնել append() մեթոդով
3. Tuple-ը չի կարող ստեղծվել դատարկ ձևով
4. Tuple-ը **չի կարող** պարունակել գրաֆիկական օբյեկտներ:
5. Tuple-ը կարող է ներառել թվեր, տողեր, ցուցակներ և Python-ի օբյեկտ չհանդիսացող տեղեկատվություն
6. Tuple-ը չի աջակցում slicing գործողություններին
7. Tuple-ները չեն կարող փոփոխվել (immutable) են
8. Tuple-ի տարրերը կարելի է հասանելի դարձնել ինդեքսի միջոցով
9. Tuple-ի տարրերը ավտոմատ դասավորվում են աճման կարգով

**Python-ի Encapsulation թեմայի վերաբերյալ տրված պնդումներից 3-ը ճիշտ են:**  
**Նշել դրանց համարները: (3 միավոր)**

1. Encapsulation-ը տվյալների և մեթոդների միավորում է մեկ կլասի մեջ
2. Encapsulation-ը պաշտպանում է օբյեկտի ներքին տվյալները արտաքին փոփոխություններից
3. Encapsulation-ը՝ ազատ մուտք ապահովելով օբյեկտին, որոշ չափով նվազեցնում է ծրագրի անվտանգության մակարդակը
4. Encapsulation-ի նպատակն է թույլ տալ ազատ մուտք օբյեկտի բոլոր ատրիբուտներին
5. Python-ում մասնավոր (private) ատրիբուտները սովորաբար սկսվում են երկու ստորագծով՝ \_\_
6. Encapsulation-ը կիրառվում է նոր կլաս ստեղծելու համար
7. Encapsulation-ը կարելի է կիրառել միայն ֆունկցիաների համար
8. Encapsulation-ը հակասում է ժառանգման գաղափարին
9. Encapsulation-ը նշանակում է տվյալների ամբողջական բացահայտում ծրագրի բոլոր մասերից

**Python-ի Set-ի վերաբերյալ տրված պնդումներից 3-ը ճիշտ են:**  
**Նշել դրանց համարները: (3 միավոր)**

1. Set-ի տարրերը միշտ դասավորվում են աճման կարգով
2. Set-ը չի աջակցում տարբերություն (difference) գործողությանը
3. Set-ի մեջ կարելի է ավելացնել list կամ dictionary որպես տարր
4. Set-ը չի կարող պարունակել անփոփոխ (immutable) տարրեր
5. Set-ի տարրերի հերթականությունը չի պահպանվում (unordered)
6. Set-ի տարրերին կարելի է հասնել ըստ ինդեքսի
7. Set-ը պահպանում է միայն եզակի (unique) տարրեր
8. Set-ը կարելի է ստեղծել փակագծերով՝ ()
9. Set-ը կարելի է միավորել մեկ այլ set-ի հետ union() կամ | օպերատորով

Նշել դրանց համարները: (3 միավոր)

1. Տողերը հնարավոր է մասամբ փոփոխել՝ որոշ նիշ փոխելով
2. Տողերի համեմատությունը (>, <, ==) կատարվում է բառարանային (lexicographical) կարգով
3. Տողերը հնարավոր չէ միացնել + օպերատորով
4. Տողում հնարավոր չէ փնտրել ենթատող in օպերատորով
5. Տողերը անփոփոխ (immutable) օբյեկտներ են
6. Տողերը չեն աջակցում slicing գործողություններին
7. Տողերը չեն կարող պարունակել escape նիշեր (օր. \n, \t)
8. Տողի երկարությունը որոշվում է count() մեթոդով
9. Տողերը կարելի է բազմապատկել ("Hi" \* 3 → "HiHiHi")

ճիշտ են: Նշել դրանց համարները: (3 միավոր)

1. Ֆունկցիաները սահմանվում են def բանալի բառով
2. Ֆունկցիան կարող է վերադարձնել արժեք return հրամանով
3. Ֆունկցիան չի կարող կանչել ինքն իրեն (recursion)
4. Բոլոր ֆունկցիաները պետք է պարտադիր վերադարձնեն արժեք
5. Ֆունկցիաները չեն կարող ունենալ լռելյայն (default) արժեքներ
6. Ֆունկցիայի անունը չի կարող պարունակել թվեր
7. Ֆունկցիան կարող է կանչվել բազմիցս՝ տարբեր արգումենտներով
8. Ֆունկցիան հնարավոր չէ սահմանել առանց պարամետրերի
9. Ֆունկցիան պետք է միշտ գրվի մեկ տողում

**Python-ի ֆայլային գործողությունների (File Handling) վերաբերյալ տրված պնդումներից 3-ը ճիշտ են: Նշել դրանց համարները: (3 միավոր)**

1. Ֆայլի պարունակությունը կարելի է կարդալ `read()` մեթոդով
2. Տեքստային ֆայլերում հնարավոր չէ գրել թվեր
3. `close()` մեթոդը փակելուց հետո կարելի է կրկին գրել նույն ֆայլում առանց բացելու
4. Ֆայլ բացելիս 'w' ռեժիմը պահպանում է նախորդ տվյալները
5. `with open()` կառուցվածքը չի կիրառվում գրելու ռեժիմում ֆայլ բացելու համար
6. Ֆայլ բացելու համար օգտագործվում է `open()` ֆունկցիան
7. Ֆայլը հնարավոր չէ բացել գրելու և կարդալու ռեժիմում միաժամանակ
8. Ֆայլը կարող է բացվել տարբեր ռեժիմներով՝ 'r', 'w', 'a', 'r+'
9. Ֆայլը փակելու անհրաժեշտություն չկա՝ Python-ը դա անում է ավտոմատ

**Python-ի Modules և Packages թեմաների վերաբերյալ տրված պնդումներից 3-ը ճիշտ են: Նշել դրանց համարները: (3 միավոր)**

1. Ներմուծված մոդուլը պետք է օգտագործվի անմիջապես բացելուց հետո, հակառակ դեպքում ջնջվում է հիշողությունից
2. Նույն մոդուլը հնարավոր չէ ներմուծել մեկ ծրագրում մեկից ավել անգամ
3. Մոդուլների ֆայլերի ընդլայնումը միշտ պետք է լինի .pkg
4. Փաթեթը (package) ֆայլերի ժողովածու է, որը պարտադիր պետք է պարունակի `__init__.py` ֆայլ
5. Մոդուլը չի կարող ներմուծվել մեկ այլ մոդուլում
6. Փաթեթները չեն կարող ունենալ ենթափաթեթներ
7. Մոդուլները չեն կարող պահվել առանձին ֆայլերում
8. Մոդուլը Python ֆայլ է, որը պարունակում է ֆունկցիաներ, դասեր կամ փոփոխականներ
9. Մոդուլ ներմուծելու համար օգտագործվում է `import` հրամանը

**Python-ի բացառությունների (Exceptions) վերաբերյալ տրված պնդումներից 3-ը ճիշտ են: Նշել դրանց համարները: (3 միավոր)**

1. except բլոկը կատարվում է միայն այն դեպքում, երբ սխալ է առաջանում
2. Բացառությունները հնարավոր չէ ժառանգել դասերից
3. Բացառությունների մշակման հիմնական նպատակը ծրագրի դադարեցումն է
4. Բացառությունները (exceptions) օգտագործվում են ծրագրի սխալները վերահսկելու և մշակելու համար
5. try բլոկում գրվում է այն կոդը, որը կարող է առաջացնել սխալ
6. finally բլոկը կատարվում է միայն այն դեպքում, երբ սխալ է առաջանում
7. Բացառությունների մշակումը դանդաղեցնում է ծրագրի աշխատանքը և չի կիրառվում պրոֆեսիոնալ կոդում
8. Միևնույն try բլոկի համար չի կարելի ունենալ մի քանի except բլոկ
9. raise հրամանով հնարավոր չէ ստեղծել սեփական բացառություններ

**Տրվածներից ո՞ր 3-ն են ճիշտ կերպով նկարագրում RAG (Retrieval-Augmented Generation) համակարգի աշխատանքի սկզբունքը և բաղադրիչները: Նշել դրանց համարները: (3 միավոր)**

1. Retrieval բաղադրիչը պատասխան է գեներացնում՝ առանց Large Language Model-ի մասնակցության:
2. RAG համակարգերում տեքստային որոնումը կատարվում է միայն հիմնաբառերով՝ առանց embeddings-ի:
3. RAG-ը թույլ է տալիս օգտագործել սեփական գիտելիքների բազա՝ առանց մոդելի վերաուսուցման:
4. RAG համակարգը համադրում է որոնման (retrieval) և գեներացիայի (generation) մեխանիզմները՝ ապահովելով պատասխանների առավել ճշգրիտ և փաստային բնույթ:
5. RAG մոդելը գործում է բացառապես առանց արտաքին տվյալների բազայի:
6. RAG-ի retrieval բաղադրիչը որոնում է համապատասխան փաստաթղթեր վեկտորային ինդեքսում (vector store):
7. Generation մոդուլը միշտ օգտագործում է fine-tuned GPT մոդել:
8. RAG համակարգերը չեն կարող օգտագործվել API միջավայրերում:
9. Վեկտորային ներկայացումները (embeddings) չեն օգտագործվում տեքստերի նմանությունը հաշվարկելու համար:

Python-ի Polymorphism թեմայի վերաբերյալ տրված պնդումներից 3-ը ճիշտ են:  
Նշել դրանց համարները: (3 միավոր)

1. Polymorphism-ը աշխատում է միայն թվային տվյալների տիպի համար
2. Եթե երկու կլասս ունեն նույն մեթոդի անունը, Python-ը սխալ է ցույց տալիս
3. Polymorphism-ը թույլ է տալիս մեթոդների տարբեր անվանումներ օգտագործել նույն նպատակով
4. Polymorphism-ը տարբեր կլասսների օբյեկտներին թույլ է տալիս նույն մեթոդը օգտագործել տարբեր ձևերով
5. Polymorphism-ը նշանակում է, որ բոլոր կլասսները պետք է ունենան իրենց մեթոդների անունները
6. Polymorphism-ը նպաստում է ծրագրի ճկունության և ընդարձակելիության բարձրացմանը
7. Polymorphism-ը գործում է միայն ժառանգված կլասսների միջև
8. Python-ում polymorphism-ը կարելի է իրականացնել միայն @staticmethod-ներով
9. Միևնույն մեթոդը չի կարող ունենալ տարբեր իրացումներ տարբեր կլասսներում